What is claimed is:

1. In a packet processor having a local packet memory (LPM) for storing packet data during processing, the LPM having at least one memory access port, a system for managing port contention, comprising:

a buffer for queueing read/write requests to the port; and

a logic mechanism associated with the buffer for determining busy status of the port for a pending request to the cell, and for issuing an appropriate command if the port is busy;

characterized in that the logic mechanism, if the port is determined to be busy, issues a command for a temporary cessation of write requests to the buffer.

2. The system of claim 1 for preventing download of packet data from the LPM before all write requests to the data are completed , wherein, if the port is busy, the logic mechanism checks the queue fir write requests, and finding a write request, also issues a command to temporarily suspend packet downloads from the LPM until all write requests in the buffer are accomplished.

3. The system of claim 1 wherein the LPM comprises a set of individual memory cells, each cell having an access port.

4. The system of claim 3 wherein the LPM has eight cells, and each cell has two memory access ports.

5. The system of claim 3 wherein packet downloading is managed by a packet management unit (PMU) issuing read requests, and read/write requests sent to the buffer are other than read requests from the PMU,

the PMU reads and the buffer sharing the same port with the PMU reads having priority, and wherein, for packets having data extensive enough to occupy two or more lines of a cell, packet data is interleaved among cells, such that consecutive cells are dedicated to different packets, ensuring that port contention is limited to alternate cycles.

6. The system of claim 3 wherein packets in process are assigned each a specific identifier, and wherein write requests in the buffer are tagged with packet identifiers, and the logic mechanism issues commands along with the packet id tag.

7. In a packet processor having a local packet memory (LPM) for storing packet data during processing, the LPM having at least one access port, a method for managing contention at a cell port, comprising the steps of:

(a) queueing read/write requests to the port in a buffer; and

(b) determining, by a logic mechanism associated with the buffer, busy status of the port for a pending request to the cell, and issuing an appropriate command for a temporary cessation of write requests to the buffer if the port is busy.

8. The method of claim 7 for preventing download of packet data from the LPM before all write requests to the data are completed , wherein, in step (b), the logic mechanism checks the buffer for write requests, and finding a write request, also issues a command to temporarily suspend packet downloads from the LPM until all write requests in the buffer are accomplished.

9. The method of claim 7 wherein the LPM comprises a set of individual memory cells, each cell having an access port.

10. The method of claim 9 wherein the LPM has eight cells, and each cell has two memory access ports.

11. The method of claim 9 wherein packet downloading is managed by a packet management unit (PMU) issuing read requests, and read/write requests sent to the buffer are other than read requests from the PMU, the PMU reads and the buffer sharing the same port with the PMU reads having priority, and wherein, for packets having data extensive enough to occupy two or more lines of a cell, packet data is interleaved among cells, such that consecutive cells are dedicated to different packets, ensuring that port contention is limited to alternate cycles.

12. The method of claim 9 wherein packets in process are assigned each a specific identifier, and wherein write requests in the buffer are tagged with packet identifiers, and the logic mechanism in step (b) issues commands along with the packet id tag.